

## TreeSolve (Version 1.0)

<https://compbio.engr.uconn.edu/software/treesolve/>

### Description

TreeSolve (portmanteau of "Tree" and "Resolve") is a program for gene tree error-correction. By default, TreeSolve is designed for the error-correction of microbial gene trees (with horizontal gene transfer) but can be easily applied to non-microbial gene trees as well. TreeSolve takes as input a rooted gene tree topology, a known rooted species tree, and a collection of (unrooted) gene tree samples such as bootstrap replicates or samples from a posterior distribution, and outputs an error-corrected gene tree topology. TreeSolve works by computing branch support values for the given rooted gene tree based on the given replicates/samples, collapsing weakly supported branches in the input gene tree, and then optimally resolving it based on both the input gene tree samples and the species tree while accounting for horizontal gene transfer, gene duplication, and gene loss. A key feature of TreeSolve is that it can output multiple distinct optimal resolutions, ordered by average branch support values. Further technical details appear in the paper cited below.

TreeSolve serves a similar purpose as TreeFix-DTL (available from <https://www.cs.hmc.edu/~yjiw/software/treefix-dtl/>), but is much faster and far more scalable while nearly matching TreeFix-DTL in accuracy, and yields multiple candidate error-corrected gene trees (rather than just a single one).

TreeSolve can be cited as follows:

#### [TreeSolve: Rapid Error-Correction of Microbial Gene Trees](#)

Misagh Kordi and Mukul S. Bansal

*Algorithms for Computational Biology (AlCoB) 2020*: LNCS; to appear

TreeSolve is freely available from <https://compbio.engr.uconn.edu/software/treesolve/>.

### Usage

TreeSolve takes as input a rooted gene tree, a rooted species tree, and a gene tree samples file containing unrooted bootstrap replicates or posterior samples for the gene tree. All trees must be provided in Newick format, and leaf labels in the gene tree and sample trees should be of the form "<SpeciesName>\_<GeneName>"; this enables mapping of the leaves of the gene tree to the leaves of the species tree. The input gene tree and gene tree samples (e.g., bootstrap replicates) can be easily constructed using standard phylogeny construction tools such as RAxML. The input gene tree can be rooted using standard rooting methods, or by using reconciliation-based rooting such as the OptRoot method available in the RANGER-DTL software package. The gene tree samples or bootstrap replicates should remain unrooted, and we recommend computing and using 100 such samples or bootstrap replicates.

In addition, users should also specify a support cutoff value, for collapsing weakly supported branches, using the -B command line option. If a support value cutoff is not specified, then a default value of 80 (%) is used.

Example input:

Gene tree file:

```
((a_p,b_q),(a_r,b_s)),(d_t,(e_u,c_v)));
```

Species tree file:

```
((a,b),((c,d),e));
```

Gene tree samples file:

```
((a_p,b_q),(a_r,b_s),(d_t,(e_u,c_v)));
```

```
((a_p,d_t),(a_r,b_s),(b_q,(e_u,c_v)));
```

```
((a_p,d_t),(b_q,(e_u,c_v)),(a_r,b_s));
```

The input gene tree and species tree should be provided in the same file using the `-i` command line option. In this file, the species tree should be on the first line and gene tree on the second line. The gene tree samples file should be specified using the `-br` command line option. An output file for writing results must also be provided using the `-o` command line option.

Many additional command line options exist, as listed below.

### List of command line options

`-i, --input` Input file name. File should contain species tree on the first line and gene tree on the second line. This is a required parameter.

`-br` File containing bootstrap replicates or other samples. This is a required parameter.

`-o, --output` Output file name. This is a required parameter.

`-D` Duplication cost (whole number only, default value 2)

`-T` Transfer cost (whole number only, default value 3)

`-L` Loss cost (whole number only, default value 1)

`-B` Bootstrap cutoff value (whole number between 0 and 100 only, default 80%)

`-N` Maximum number of optimal resolutions to be printed to output file (default value 25000). If the actual number of optimal reconciliations is greater than this value, then the specified number of resolutions is output sorted by average bootstrap support; e.g., the first tree output is the “best” optimal resolution found.

`-nb` Number of bootstrap replicates to use (default value 100).

`-ts` Maximum size of table to be used (default value 25000).

`-s, --summary` Only output summary statistics

- q, --quiet      Suppress program output. Only output optimized reconciliation cost, number of optimal resolutions, and list of optimal resolutions.
- v, --version    Version number
- h, --help      Brief help message

### Interpretation of the output

The output from an execution of TreeSolve is written to the specified output file. This output consists of basic statistics for the input gene tree (including maximum out-degree, number of non-binary nodes, and current reconciliation cost), followed by the optimal DTL reconciliation cost (obtained after error-correcting with TreeSolve) and the number of optimal resolutions of the gene tree, and finally a list of all optimal resolutions of the gene tree ordered by average branch support (bootstrap or other).

### Example datasets

The “TestData” directory contains a sample input file (containing species tree and gene tree) and a corresponding bootstrap replicates file that can be used as input for TreeSolve. TreeSolve can be applied to this test data as follows:

```
./TreeSolve.linux -i Tree28.newick -br RAxML_bootstrap.G28 -o Tree28.Output
```

### Contact Information

In case of any questions, please feel free to contact [Mukul Bansal](#) at [mukul.bansal@uconn.edu](mailto:mukul.bansal@uconn.edu).